2020 SISBID High-Dimensional Hypothesis Testing Lab

Genevera I. Allen, Yufeng Liu, Hui Shen, Camille Little

7/21/2020

Load Packages

```
library(sda)
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.6.2
```

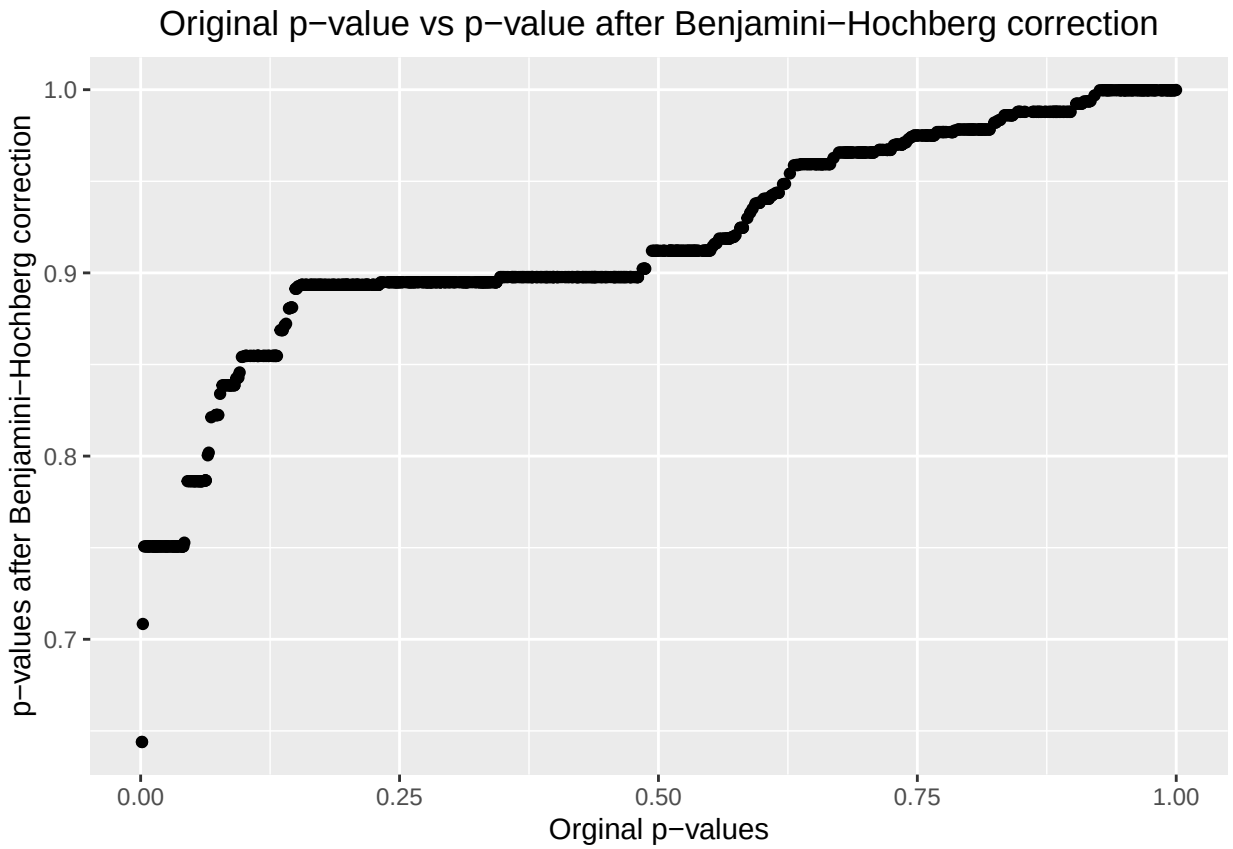
H_0 : feature is not associated with the response.
Data set 1 - Simulated Data
Small simulated data set to demonstrate multiple testing when **all null hypthesis hold**.

```
#simulate data
x <- matrix(rnorm(1000*50),ncol=50)
y <- sample(c(0,1),50,rep=TRUE)
ps <- NULL
for(i in 1:1000){
  ps <- c(ps,t.test(x[i,y==0],x[i,y==1])$p.value)
}
cat("Around 5% of p-values are below 0.05:",mean(ps<.05),fill=TRUE)
```

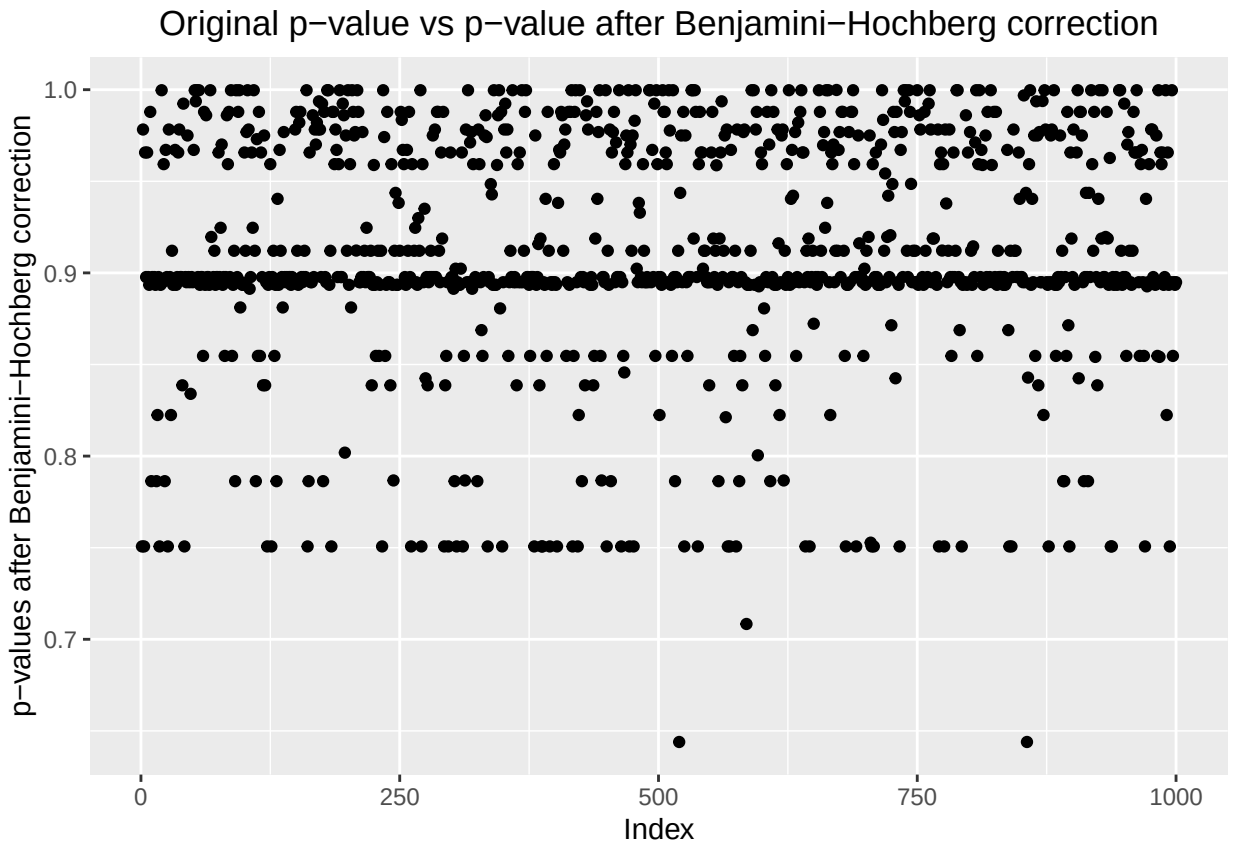
```
## Around 5% of p-values are below 0.05: 0.061
```

Benjamini-Hochberg Algorithm for FDR Control

```
fdrs.bh <- p.adjust(ps, method="BH")
BHData = data.frame(cbind(ps,fdrs.bh))
colnames(BHData) = c("OriginalP","BH.P")
ggplot(BHData) +
  geom_point(mapping = aes(x = OriginalP, y = BH.P)) +
  ggtitle("Original p-value vs p-value after Benjamini-Hochberg correction") +
  theme(plot.title = element_text(hjust = 0.5)) +
  xlab("Orginal p-values") + ylab("p-values after Benjamini-Hochberg correction")
```




```
BHData$index = 1:nrow(BHData)
ggplot(BHData) +
  geom_point(mapping = aes(x = index, y = BH.P)) +
  ggtitle("Original p-value vs p-value after Benjamini-Hochberg correction") +
  theme(plot.title = element_text(hjust = 0.5)) +
  xlab("Index") + ylab("p-values after Benjamini-Hochberg correction")
```



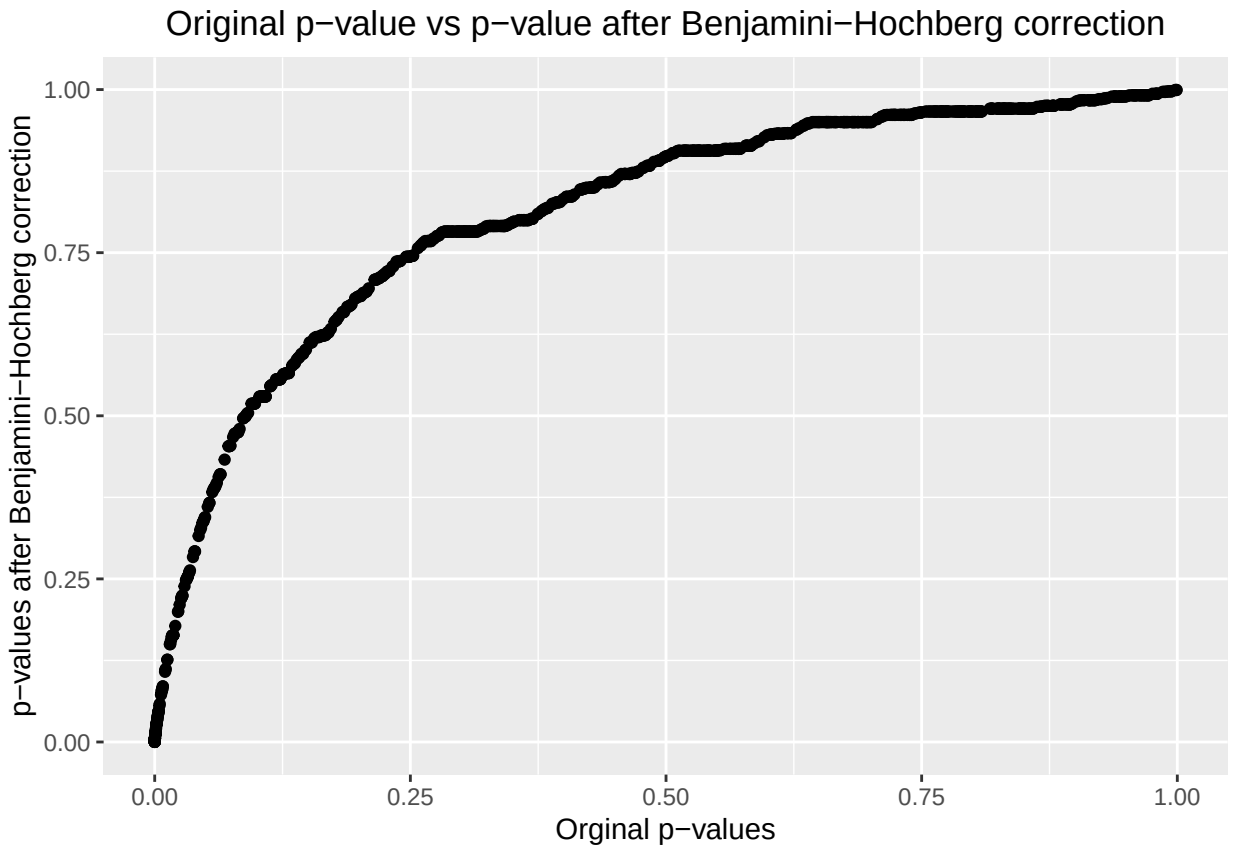
Data set 2 - Simulated Data

Small simulated data set to demonstrate multiple testing when **not all null hypthesis hold**.

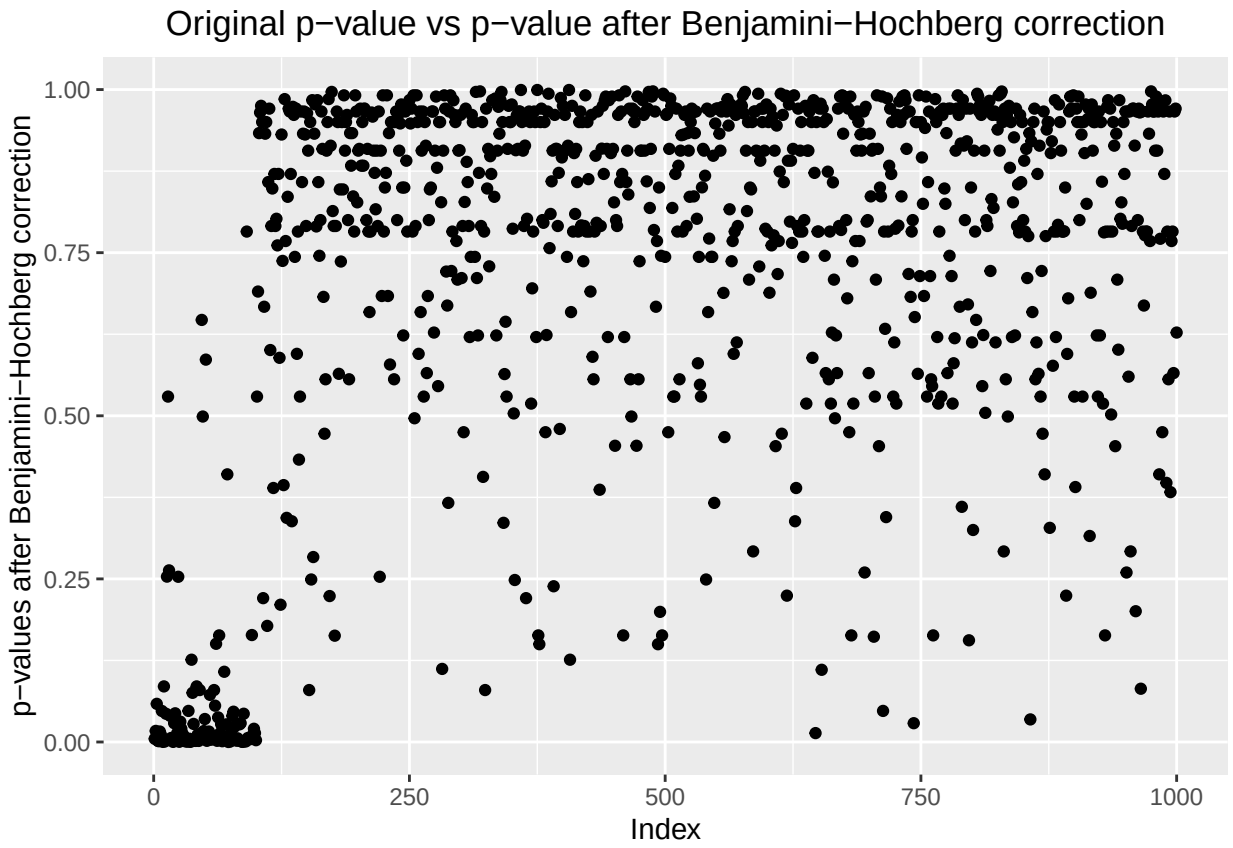
```
#simulate data
x <- matrix(rnorm(1000*50),ncol=50)
y <- sample(c(0,1),50,rep=TRUE)
x[1:100,y==0] <- x[1:100,y==0] + 1
ps <- NULL
for(i in 1:1000) {
  ps <- c(ps,t.test(x[i,y==0],x[i,y==1])$p.value)
}
cat("Way more than 5% of p-values are below 0.05:",mean(ps<.05),fill=TRUE)
```

```
## Way more than 5% of p-values are below 0.05: 0.143
```

```
fdrs.bh <- p.adjust(ps, method="BH")
# plot
BHData = data.frame(cbind(ps,fdrs.bh))
colnames(BHData) = c("OriginalP","BH.P")
ggplot(BHData) +
  geom_point(mapping = aes(x = OriginalP, y = BH.P)) +
  ggtitle("Original p-value vs p-value after Benjamini-Hochberg correction") +
  theme(plot.title = element_text(hjust = 0.5)) +
  xlab("Orginal p-values") + ylab("p-values after Benjamini-Hochberg correction")
```




```
BHData$index = 1:nrow(BHData)
ggplot(BHData) +
  geom_point(mapping = aes(x = index, y = BH.P)) +
  ggtitle("Original p-value vs p-value after Benjamini-Hochberg correction") +
  theme(plot.title = element_text(hjust = 0.5)) +
  xlab("Index") + ylab("p-values after Benjamini-Hochberg correction")
```



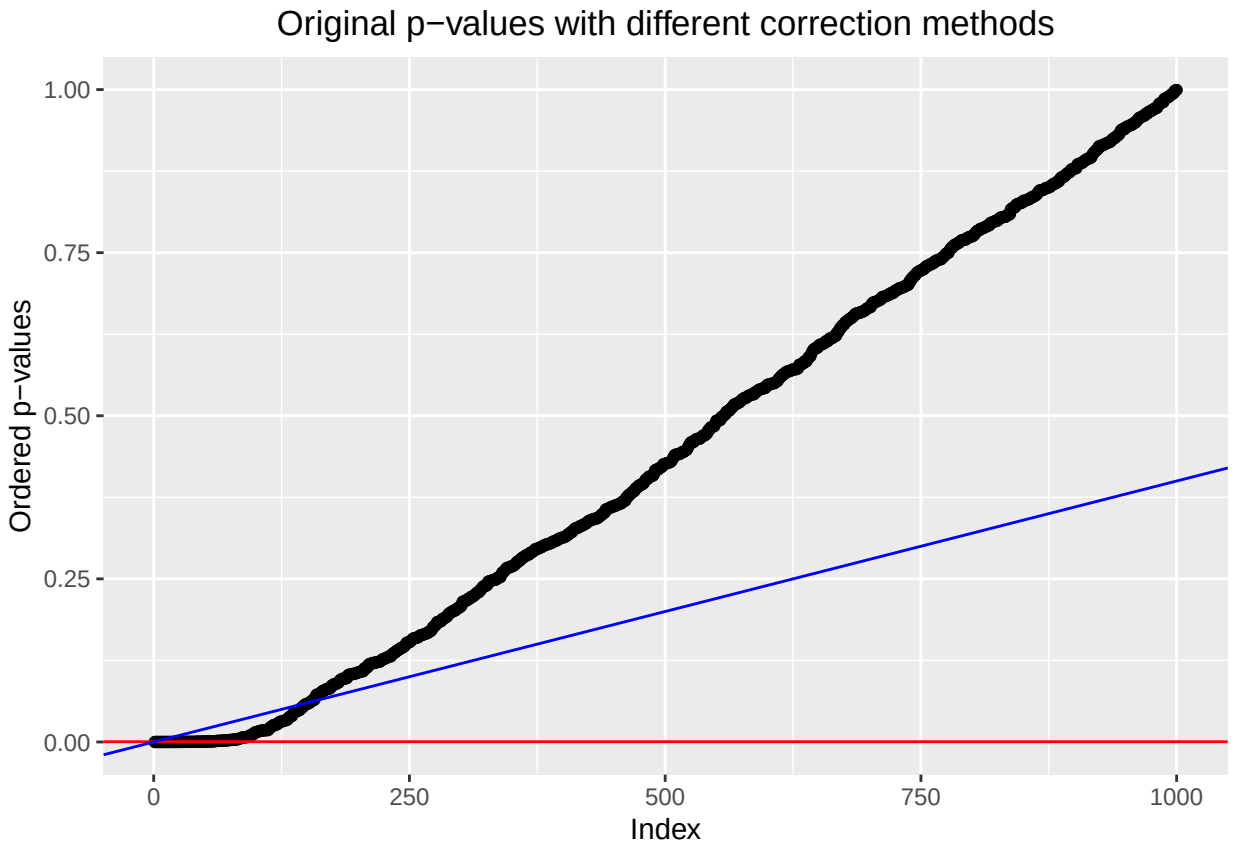

```
cat("Number of Tests with FDR below 0.4:",sum(fdrs.bh<0.4), fill=TRUE)
```

```
## Number of Tests with FDR below 0.4: 153
```

```
cat("Compute the BH FDR Directly:",max(which(sort(ps,decreasing=FALSE) < .4*(1:1000)/1000)),
                                    fill=TRUE)
```

```
## Compute the BH FDR Directly: 153
```

```
BHData = BHData[order(ps,decreasing = FALSE),]
BHData$index = 1:nrow(BHData)
# plot
ggplot(BHData) +
  geom_point(mapping = aes(x = index, y = OriginalP)) +
  ggtitle("Original p-values with different correction methods") +
  geom_abline(intercept = 0.4/1000,slope = 0,col= "red") + #Bonferroni
  geom_abline(intercept = 0 ,slope = 0.4/1000,col= "blue") + #BH procedure
  theme(plot.title = element_text(hjust = 0.5)) +
  xlab("Index") + ylab("Ordered p-values")
```



Data set 3, Real Data: Prostate Data (Singh et al. 2002). This data set consists of gene expression levels for 6033 genes among 102 men.

The dataset is available from the R package "sda"
* Problem 1 - We wish to identify important genes to differetiate cancer or healthy patients. What kind of tests are reasonable?
* Problem 2 - In order to adjust for multiple comparisons, which procedures should one use?
* Problem 3 - Examine the list of genes identified.

```
## import data
data(singh2002)
x = singh2002$x
y = singh2002$y

n1 = sum(y == "healthy")
n2 = length(y) - n1
```

```
ps<-NULL
for(i in 1:ncol(x)) {
  ps <- c(ps, t.test(x[1:n1,i], x[(n1+1):(n1+n2),i])$p.value)
}
## ordered p-values names(ps)<-seq(1,ncol(x),1)
p1 =sort (ps)

## plot ordered p-values
plot(p1[1:100], pch=rep('*',100),ylim=c(0,0.003), ylab="ordered p-values")
## rejection boundry of Benjamini-Hochberg's procedure
abline(a=0, b=0.1/ncol(x), col="red")
```

```r
## rejection boundary of Bonferroni at 0.1
abline(a=0.1/ncol(x), b=0, col="blue",lty=5)

cat("Compute the no. rejection by Bonferroni:",
    max(which(sort(ps,decreasing=FALSE) < .1/ncol(x))), fill=TRUE)
```

```
## Compute the no. rejection by Bonferroni: 6
```

```r
cat("Compute the BH FDR Directly:",
max(which(sort(ps,decreasing=FALSE) < .1*(1:ncol(x))/ncol(x))),
                                        fill=TRUE)
```

```
## Compute the BH FDR Directly: 57
```

```r
arrows(x0 = 61, y0 = 0.00085, x1 = 58, y1 = p1[57], length = 0.1)
text(63.5, 0.00085, labels="imax = 57", cex=.8, pos=4, col="black")
legend("topleft",legend=c("BH's Procedure", "Bonferroni", "Ordered p-values"),
       lty=c(1, 5, NA), col=c("red","blue", "black"), pch = c(NA, NA, '*'))
```